Computably totally disconnected, locally compact groups

André Nies
Joint work with Alexander Melnikov

Cross-Alps Logic Seminars, June 2023

THE UNIVERSITY OF AUCKLAND
NEW ZEALAND

Polish groups, non-Archimedean groups

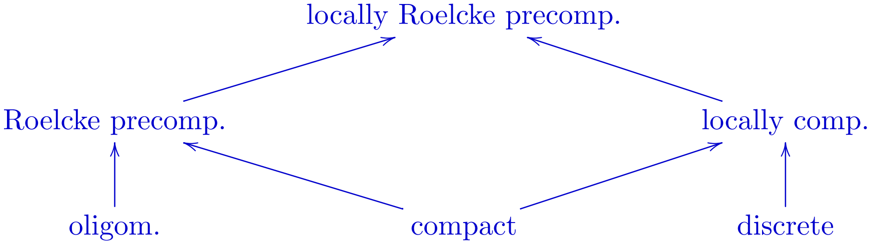
Topological group: a group with a topology such that the operations are continuous. Note that each open subgroup is closed.

Polish group: the topology is complete metrizable, separable.

Non-Archimedean group: there is basis of the identity consisting of (cl)open subgroups. Such groups are totally disconnected.

They are, up to homeomorphism, the closed subgroups of the topological group $\mathrm{Sym}(\mathbb{N})$ of permutations of $\mathbb{N}$ with the usual topology of pointwise convergence. In other words, they are the automorphism groups of structures with domain $\mathbb{N}$.

Borel classes of closed subgroups of S_∞



Arrows denote inclusion of the classes. More detail on Logic Blog 2022.

- $\cong$ on the profinite groups is $\equiv_B$ graph isomorphism. Kechris, N. and Tent (JSL, 2018)
- $\cong$ on the class of oligomorphic groups is $\leq_B$ a countable Borel equivalence relation. N, Schlicht and Tent (JML, 2021)

Brief intro to totally disconnected,

locally compact (t.d.l.c.) groups

To say that a (separable) topological group G is t.d.l.c. means just what it says: G is locally compact and totally disconnected (the clopen sets form a basis). This implies Polish.

Van Dantzig's theorem (1936): Each t.d.l.c. group G has a basis of neighbourhoods of 1 consisting of compact open subgroups.

In particular, G is non-Archimedean.

Examples of t.d.l.c. groups G

- All computable profinite groups and all computable discrete groups. G, resp $\{1\}$, is a compact open subgroup.

- $(\mathbb{Q}_p, +)$, the additive group of p-adic numbers for a prime p. $\mathbb{Z}_p$ is a compact open subgroup.

- The semidirect product $\mathbb{Z} \ltimes \mathbb{Q}_p$ where $g \in \mathbb{Z}$ acts as $x \mapsto xp$ on $\mathbb{Q}_p$. And $\mathbb{Z}_p$ is a compact open subgroup.

- The groups $\mathrm{SL}_n(\mathbb{Q}_p)$ for $n \geq 2$.
  Here $\mathrm{SL}_n(\mathbb{Z}_p)$ is a compact open subgroup.

- $Aut(T_d)$, the automorphisms of a homogeneous undirected tree of degree d. More generally, automorphism groups of countable locally finite graphs. Stabilizer of a vertex is a compact open subgroup.

The main questions

(A) How can one define a computable presentation of a t.d.l.c. group? Which t.d.l.c. groups have such a presentation?

(B) Given a computable presentation of a t.d.l.c. group, are objects such as the (rational-valued) Haar measures, the modular function, or the scale function computable?

(C) Do constructions that lead from t.d.l.c. groups to new t.d.l.c. groups have algorithmic versions?

Question A: How can one define a computable presentation of a t.d.l.c. group?

- We will introduce two notions of computable presentation of a t.d.l.c. group G, and show their equivalence.
- The first notion relies on standard notions of computability in the uncountable setting.
- The second notion works with computation on a countable structure of approximations of the elements, the "meet groupoid" of G. Its domain is given by the compact open cosets of G.
- All the examples above, such as $(\mathbb{Q}_p, +)$, $\mathrm{SL}_2(\mathbb{Q}_p)$ and $\mathrm{Aut}(T_d)$, have computable presentations.

Questions B and C

- We show that given a computable presentation of a t.d.l.c. group G, the modular function and the Cayley-Abels graphs (in the compactly generated case) are computable.
- We discuss algorithmic properties of the scale function on G, and build a G where it is not computable.
- We explain why the class of computably t.d.l.c. groups is closed under most of the constructions studied by Wesolek (PLMS, 2015).

Computable structures: the countable case

Definition (Mal'cev and Rabin independently, 1960s)

- A computable structure is a structure such that the domain is a computable set $D \subseteq \mathbb{N}$, and the functions and relations of the structure are computable.
- A countable structure S is called computably presentable if some computable structure W is isomorphic to it.
- One says that W is a computable copy of S.

For instance, for each $k \geq 1$, the group $GL_k(\mathbb{Q})$ has a computable copy.

Computable structures: the uncountable case

- One represents all the elements by "names", which are directly accessible to computation of Turing machines with tapes that hold the infinite inputs.
- Let $\mathbb{N}^*$ denote the tree of strings with natural number entries. Names usually are elements of the set $[T]$ of paths on some computable subtree T of $\mathbb{N}^*$.
- One can now define computability of functions and relations on $[T]$: one requires that they are computable on the names.
- For instance, as a name for a real number r, take a path denoting a sequence of rationals $\langle q_n \rangle_{n \in \mathbb{N}}$ such that

  $$\forall n \, |q_n - q_{n+1}| \leq 2^{-n} \text{ and } \lim_n q_n = r.$$

  The structure $(\mathbb{R}, +\times, \exp)$ is now computable in this sense.

Ad hoc definition of computability for a profinite group

Smith, and la Roche independently, in papers dating both from 1981, called a profinite group G computable if

$$G = \varprojlim_i (A_i, \psi_i)$$

for a computable diagram $(A_i, \psi_i)_{i \in \mathbb{N}}$ of finite groups and epimorphisms $\psi_i \colon A_i \to A_{i-1}$ $(i > 0)$.

Computable presentations of t.d.l.c. groups

- We aim at a robust definition of when a t.d.l.c. group has a computable presentation.
- We ask that our definition extend the existing definitions for discrete, and for profinite groups.
- We want this class to have good algorithmic closure properties.

We provide two types of computable presentations, which will turn out to be equivalent in the sense that from a presentation of one type one can construct a presentation of the other type.

1. Defining computably t.d.l.c. groups

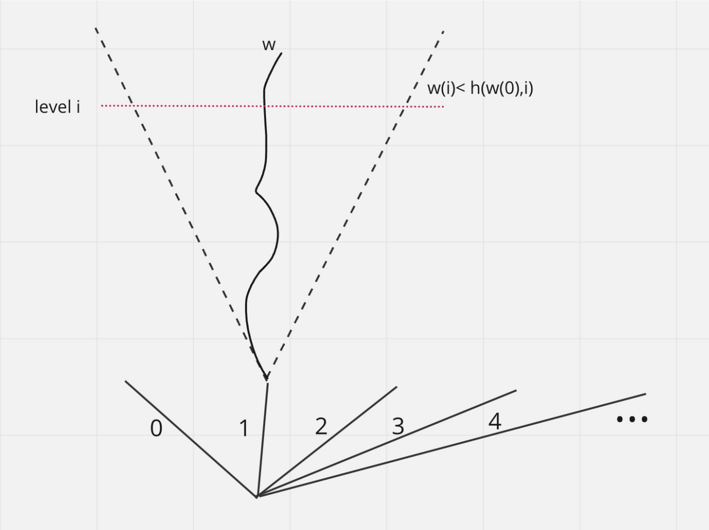
via Baire presentation

Computable Baire presentation of t.d.l.c. G

We use that each totally disconnected Polish space is homeomorphic to the set of paths $[T]$ for some subtree T of $\mathbb{N}^*$. So "names= elements".

- The domain of G equals $[T]$ for a "computably locally compact" subtree of $\mathbb{N}^*$: T is computable.
  - The only possible infinite branching is at the root.
  - There is a computable bound $h\colon \mathbb{N} \to \mathbb{N}$ such that $w(i) \le h(i, w(0))$ for each $w \in T$ and $i > 0$. (That is, the tree above n is finitely branching, effectively in n.)
- The operations are computable: A Turing machine holds the infinite inputs on tapes which keep unchanged during the computation. It can determine any number of symbols of the output from sufficiently many queries to the input tapes.

Computably locally compact tree



The ring $\mathbb{Q}_p$ has a computable Baire presentation

Let Q be the tree of strings $\sigma \in \mathbb{N}^*$ such that

- all entries, except possibly the first, are among $\{0, \ldots, p-1\}$,
- $r0 \not\preceq \sigma$ for each $r > 0$.

A string $r\sigma \in Q$ denotes the rational $p^{-r} n_\sigma \in \mathbb{Z}[1/p]$, where

$$n_\sigma = \textstyle\sum_{i<|\sigma|} p^i \sigma(i).$$

The condition that $r0 \not\preceq \sigma$ for each $r > 0$ says that p does not divide n_σ. (Else we'd have a better notation of the same rational, starting with $r-1$.)

One checks that addition and multiplication on $\mathbb{Q}_p$ are computable in the sense of infinite tapes for inputs and output.

2. Defining computably t.d.l.c. groups

via meet groupoids

The meet groupoid

Recall van Dantzig: there's a basis of neighbourhoods of 1 consisting of compact open subgroups.

- We introduce an algebraic structure $\mathcal{W}(G)$ on the countable set of compact open cosets in G, together with $\emptyset$.
- This is a natural algebraic structure on approximations to elements of G.
- This structure is a partially ordered groupoid, with the usual set inclusion, and multiplication of a left coset of a subgroup U with a right coset of U (which results in a coset).
- The intersection of two compact open cosets is such a coset itself, unless it is empty, so we have a meet semilattice.

What's a groupoid? (Old notion)

Intuitively, the notion of a groupoid generalizes the notion of a group by allowing that the binary operation is partial.

- A groupoid is given by a domain $\mathcal{W}$ on which a unary operation $(.)^{-1}$ and a partial binary operation, denoted by "$\cdot$", are defined.
- Category view: a groupoid is a small category in which each morphism has an inverse.
- $A\colon U \to V$ means that U, V are idempotent ($U \cdot U = U$), and $A = UA = AV$.

What's a meet groupoid? (New notion)

Definition

A meet groupoid is a groupoid $(\mathcal{W}, \cdot, (.)^{-1})$ that is also a meet semilattice $(\mathcal{W}, \cap, \emptyset)$ of which $\emptyset$ is the least element.
Writing $A \subseteq B \iff A \cap B = A$, it satisfies the conditions

- $\emptyset^{-1} = \emptyset = \emptyset \cdot \emptyset$, and $\emptyset \cdot A$ and $A \cdot \emptyset$ are undefined for each $A \neq \emptyset$,
- if U, V are idempotents such that $U, V \neq \emptyset$, then $U \cap V \neq \emptyset$,
- $A \subseteq B \iff A^{-1} \subseteq B^{-1}$, and
- if $A_i \cdot B_i$ are defined $(i = 0, 1)$ and $A_0 \cap A_1 \neq \emptyset \neq B_0 \cap B_1$, then

$$(A_0 \cap A_1) \cdot (B_0 \cap B_1) = A_0 \cdot B_0 \cap A_1 \cdot B_1.$$

The meet groupoid of a t.d.l.c. group

Definition

Let G be a t.d.l.c. group. We define a meet groupoid $\mathcal{W} = \mathcal{W}(G)$.

- Its domain consists of the compact open cosets in G, as well as the empty set. Meet is intersection.
- We define $A \cdot B$ to be the usual product AB in case that $A = B = \emptyset$, or A is a left coset of a subgroup V and B is a right coset of V; otherwise $A \cdot B$ is undefined.

Proposition (N., Logic Blogs '22, '23)

- $Aut(G)$ (Braconnier topology) is homeomorphic to $Aut(\mathcal{W})$.
- The Chabauty space $\mathcal{S}(G)$ of closed subgroups of G can be canonically represented by a closed subset of $2^{\mathcal{W}}$, consisting of certain ideals of $\mathcal{W}$.

Computably t.d.l.c. groups via meet groupoids

A meet groupoid $\mathcal{W}$ is called Haar computable if

(a) its domain is a computable subset D of $\mathbb{N}$;

(b) the groupoid and meet operations are computable; in particular, the relation $\{\langle x, y\rangle \colon x, y \in S \wedge x \cdot y \text{ is defined}\}$ is computable;

(c) the function sending a pair of idempotents $U, V \in \mathcal{W}$ to the number of (left) cosets of $U \cap V$ in U is computable.

Definition (Computably t.d.l.c. groups via meet groupoids)

Let G be a t.d.l.c. group. We say that G is computably t.d.l.c. via a meet groupoid if $\mathcal{W}(G)$ has a Haar computable copy $\mathcal{W}$.

Computably t.d.l.c. via meet groupoid

Example (4.9)

For any prime p, the additive group $\mathbb{Q}_p$ and the group $\mathbb{Z} \ltimes \mathbb{Q}_p$ are computably t.d.l.c. via a meet groupoid.

$\boxed{\mathbb{Q}_p}$: compact open subgroups are of the form $U_r := p^r\mathbb{Z}_p$ for some $r \in \mathbb{Z}$, all compact. For each r there is a canonical epimorphism $\pi_r \colon \mathbb{Q}_p \to C_{p^\infty}$ with kernel U_r. So each compact open coset of U_r can be uniquely written in the form $D_{r,a} = \pi_r^{-1}(a)$ for some $a \in C_{p^\infty}$.

$\boxed{\mathbb{Z} \ltimes \mathbb{Q}_p}$: it has the same compact open subgroups as $\mathbb{Q}_p$.
We have $D_{r,a} = g^{-z} D_{r-z,a} g^z$ for each $z \in \mathbb{Z}$. So we have $A_{r,z} \colon U_{r-z} \to U_r$ where $A_{r,z} = g^z U_r$. In particular $A_{1,1} \colon U_0 \to U_1$.
For both groups, the meet groupoid operations and index function are computable.

Equivalence of

the two types of computable presentations

Equivalence of the two kinds of computable presentations

Theorem

A group G is computably t.d.l.c. via a Baire presentation $\Longleftrightarrow$ G is computably t.d.l.c. via a meet groupoid.

From a presentation of G of one type, one can uniformly obtain a presentation of G of the other type.

$\Leftarrow$: From meet groupoid to Baire presentation

Given: a Haar computable meet groupoid $\mathcal{W}$ with domain $\mathbb{N}$ such that $\mathcal{W}(G) \cong \mathcal{W}$ (identify).

Let $\mathcal{G}(\mathcal{W})$ be the closed subgroup of S_∞ consisting of elements p that preserve the meet operation of $\mathcal{W}$, and satisfy

$$p(A) \cdot B = p(A \cdot B)$$

whenever $A \cdot B$ is defined.

- We show that $\mathcal{G}(\mathcal{W}) \cong G$.
- We get a computable Baire presentation of $\mathcal{G}(\mathcal{W})$ by defining a computably locally compact tree T whose paths denote pairs of a permutation p as well as its inverse p^{-1}; the permutation satisfies the property above.

$\Rightarrow$: From Baire presentation to meet groupoid

Theorem (Recall)

A group G is computably t.d.l.c. via a Baire presentation $\iff$ G is computably t.d.l.c. via a meet groupoid.

"$\Rightarrow$":

- the domain of the computable presentation of the meet groupoid consists of codes for finite sets of nonempty strings $\sigma_1, \dots, \sigma_n$ such that the compact open set $\bigcup_i [\sigma_i]_T$ is a coset.
- Given that the Baire presentation is computable, we can compute with those sets.
- Using this we show that we obtain a Haar computable meet groupoid.

The actions on $\mathcal{W}$ are computable

Corollary (to "$\Rightarrow$" of the proof above)

- Let $\mathcal{W}$ be a Haar computable copy of $\mathcal{W}(G)$ (with domain $\mathbb{N}$).
- The left and right actions $[T] \times \mathbb{N} \to \mathbb{N}$, given by
$$(g, A) \mapsto gA \text{ and } (g, A) \mapsto Ag,$$
are computable.

Computability in the abelian case

Theorem (Lupini, Melnikov and N., J Algebra, 2022)

Let G be abelian t.d.l.c. group. The following are equivalent.

(1) G is computably t.d.l.c.

(2) There are a computable profinite group K and a computable discrete group L such that G is a topological extension of L by K via a computable co-cycle $c \colon L \times L \to K$.

(The second condition means we have an exact sequence $0 \to K \to G \to L \to 0$.)

Algorithmic properties of objects associated with a t.d.l.c. group

The modular function is computable

Throughout, let G be computably t.d.l.c. via a Baire presentation based on $[T]$, and let $\mathcal{W}$ be the Haar computable copy of $\mathcal{W}(G)$

- by definition $\Delta(g) = \mu(Ug)/\mu(U)$, where U is any compact open subgroup, μ a left Haar measure;
- we may assume μ is rational valued, and hence that μ is computable.

Since the right action of G on $\mathcal{W}$ is computable, we have:

Proposition

The modular function $\Delta\colon [T] \to \mathbb{Q}^+$ is computable.

Cayley-Abels graphs are computable

If G is compactly generated, there is a compact open subgroup U, and a set $S = \{s_1, \ldots, s_k\} \subseteq G$ such that $S = S^{-1}$ and $U \cup S$ algebraically generates G. The Cayley-Abels graph

$$\Gamma_{S,U} = (V_{S,U}, E_{S,U})$$

of G is given as follows. The vertex set $V_{S,U}$ is the set $L(U)$ of left cosets of U, and the edge relation is

$$E_{S,U} = \{\langle gU, gsU\rangle \colon\ g \in G, s \in S\}.$$

Theorem

Suppose that G is computably t.d.l.c. and compactly generated. Each Cayley-Abels graph $\Gamma_{S,U}$ of G has a computable copy $\mathcal{L}$.

Algorithmic properties of the scale function

For a compact open subgroup V of G and an element $g \in G$ let $m(g, V) = |V^g \colon V \cap V^g|$.
Define the scale function $[T] \to \mathbb{N}$ by

$$s(g) = \min\{m(g, V)\colon\ V \text{ is a compact open subgroup}\}.$$

E.g., in $\mathbb{Z} \ltimes \mathbb{Q}_p$, where $g \in \mathbb{Z}$ acts as $x \to xp$, we have $s(g) = 1$, $s(g^{-1}) = p$.

Fact

The scale function is computably approximable from above.

Noncomputable scale

For the given examples the scale is computable. However:

Theorem

There is a computable presentation of a t.d.l.c. group G based on a tree T such that the scale function $s\colon [T] \to \mathbb{N}$ is not computable.

In fact there is a uniformly computable sequence $(g_n)_{n\in\mathbb{N}}$ in G such that $s(g_n) = 2$ if $n \notin K$, and 1 otherwise.

Open question: is there is a computably presented G such that the scale is non-computable for each of its computable presentations?

Closure properties

of the class of computably t.d.l.c. groups

The class of computably t.d.l.c. groups is closed under suitable algorithmic versions of many constructions that have been studied in the theory of t.d.l.c. groups:

- passing to closed subgroups,
- taking group extensions via continuous actions,
- forming "local" direct products, and
- taking quotients by closed normal subgroups

The first three are reasonably straightforward.

Quotients by computable closed normal subgroups

The fourth isn't...

Theorem (Thm. 11.11 in paper with Melnikov)

Let N be a closed normal subgroup of G such that $\mathit{Tree}(N)$ is a computable subtree of $\mathit{Tree}(G)$. Then G/N is computably t.d.l.c.

We prove this by building a Haar computable copy of the meet groupoid of G/N.
On the way we have to show that "$\mathcal{K} \subseteq N\mathcal{L}$" is decidable, where $\mathcal{K}, \mathcal{L}$ are compact open sets.

References:

Melnikov and N. Computably t.d.l.c. groups. Arxiv preprint, 2022, https://arxiv.org/pdf/2204.09878.pdf

Lupini, Melnikov and N. Computable topological abelian groups. Journal of Algebra, Oct 2022, Arxiv: 2105.12897

N., Schlicht and Tent, Coarse groups, and the isomorphism problem for oligomorphic groups. J. Math Logic, 2021, Arxiv: 1903.08436.

Kechris, N., and Tent. The complexity of topological group isomorphism. The Journal of Symbolic Logic 83.3 (2018): 1190-1203.